

WHAT IS CLAIMED IS:

1. A method to test a code segment of a source file, comprising:
machine-rendering a source code skeleton in response to a selection of the code segment;
incorporating the code segment into the source code skeleton to generate a temporary
5 source file;
inserting a monitoring statement into the temporary source file, the monitoring statement
to provide runtime data associated with the code segment; and
compiling the temporary source file into a compiled program to output a result based
upon the monitoring statement when the temporary source file is executed.
10
2. The method of claim 1, further comprising:
executing the compiled program; and
outputting the result in response to the executing, wherein the result is based upon the
monitoring statement.
15
3. The method of claim 1, wherein the compiling comprises:
initiating compilation of the temporary source file;
attempting to resolve a compilation error; and
outputting the compilation error.
20
4. The method of claim 3, further comprising assigning data to a variable to resolve the
compilation error.
5. The method of claim 1, wherein the compiling comprises:
25 initiating compilation of the temporary source file; and
resolving a compilation error encountered during compilation.
6. The method of claim 1, wherein machine-rendering comprises copying external code
referenced by the code segment into the temporary source file.
30

7. The method of claim 1, wherein inserting the monitoring statement comprises prompting a programmer to select a variable to associate with the result and inserting an assignment statement into the temporary source file to capture the runtime data from the selected variable.

5

8. The method of claim 1, wherein inserting the monitoring statement comprises inserting a time stamp statement into the temporary source file to capture a time stamp.

9. A system to test a code segment of a source file, comprising:
a file creator to create a temporary source file in response to a selection of the code
segment, wherein the temporary source file is based upon the code segment;
a code gatherer to copy external code referenced by the code segment into the temporary
source file;
a code generator to insert a monitoring statement, the monitoring statement to provide
runtime data associated with the code segment;
an adaptive compiler to compile the temporary source file into a compiled program to
generate a result based upon the monitoring statement;
a processor to execute the compiled program; and
an output device to communicate the result.
10. The system of claim 9, wherein the file creator comprises an extensible integrated
development environment having a language development tool plug-in.
11. The system of claim 9, wherein the file creator comprises a program editor selected from
a group of program editors comprising a Graphical User Interface program editor and a
command line program editor.
12. The system of claim 9, wherein the file creator is able to create the temporary source file
based upon the code segment, wherein the code segment is selected from a group of code
segments comprising a code segment to parse strings, a code segment to perform binary
shifting, a code segment to format files, a code segment of an Application Programming
Interface, and a code segment of a library.
13. The system of claim 9, wherein the adaptive compiler is able to initiate compilation of the
temporary source file before the source file is able to be compiled.
14. The system of claim 9, wherein the adaptive compiler is able to initiate compilation of the
temporary source file, output a compilation error, and alter contents of the temporary
source file to resolve the compilation error.

15. A machine-accessible medium containing instructions, which when executed by a machine, cause said machine to perform operations, comprising:
machine-rendering a source code skeleton in response to a selection of a code segment;
incorporating the code segment into the source code skeleton to generate a temporary
5 source file;
inserting a monitoring statement into the temporary source file, the monitoring statement
to provide runtime data associated with the code segment; and
compiling the temporary source file into a compiled program to output a result based
upon the monitoring statement when the temporary source file is executed.
- 10
16. The machine-accessible medium of claim 15, further comprising:
executing the compiled program; and
outputting the result in response to the executing, wherein the result is based upon the
monitoring statement.
- 15
17. The machine-accessible medium of claim 16, further comprising terminating the
executing in response to selection of a cancel button by a user.
18. The machine-accessible medium of claim 15, wherein compiling comprises inserting a
20 line of code into the temporary source file to resolve a compilation error.
19. The machine-accessible medium of claim 15, wherein compiling comprises prompting a
user to input a library name to insert the line of code from the library into the temporary
source file to resolve a compilation error.
- 25
20. The machine-accessible medium of claim 15, further comprising deleting the temporary
source file.